Available online at www.sciencedirect.com

SciVerse ScienceDirect

Procedia Engineering 00 (2015) 000–000

Procedia Engineering

www.elsevier.com/locate/procedia

24th International Meshing Roundtable (IMR24)

Partial remeshing strategy for CFD simulations when large displacements of moving bodies are taken into account

Simone Gremmo[a,*], Iouri Antonik[b], Pascal Corso[a], Charles Hirsch[b], Grégory Coussement[a]

[a] *University of Mons - UMONS, Fluids-Machines Laboratory, 20 Place du Parc, 7000 Mons, BELGIUM*
[b] *NUMECA International, Chaussée de la Hulpe, 189, Terhulpsesteenweg, B-1170 Brussels, BELGIUM*

Abstract

This research aims at developing innovative and efficient strategies to run computational fluid dynamics simulations when large displacements of moving bodies are taken into account. The goal consists of developing new tools to follow the movement of a rigid body in a given three dimensional domain during the simulation, regardless of the displacement and the rotation amplitudes. In this paper the strategies concerning the mesh generation and update are described. This research is conducted within a collaboration between the Fluids-Machines Laboratory of the University of Mons and NUMECA International.

Peer-review under responsibility of organizing committee of the 24th International Meshing Roundtable (IMR24).

Keywords: Partial remeshing; moving body; large displacements; hybrid mesh

1. Positioning of the problem

The present research is carried out in the domains of moving bodies and fluid dynamics. Moving body simulations are used in different fields such as aeronautics and aerospace, biomedical applications, transports, etc. A typical application concerns the problem of store separation, which that aims at simulating the release of a load from a flying aeroplane [1]. Numerical simulations of such types of problems are more and more common due to the growing computational resources available [2], but multiple challenges raise in terms of both mesh generation [3] and computation [4] of the Navier-Stokes equation solution. The main concerns are related to the time required to get an accurate solution (computational time) and to the implementation of a robust and effective algorithm especially in parallel environments to be able to take advantage of the large parallel computers. In literature, different approaches to solve the rigid moving body problem can be found and they can be separated into three main categories [5], each having its own advantages and limitations: Chimera approach [6,7], mesh deformation [8,9] and remeshing [10,11]. To implement the strategy, it has been chosen to use CFD tools developed by *NUMECA International*. The mesh generator is *HEXPRESS/Hybrid* (*HH*) and it will be coupled with the solver *FINE/Open* for compressible and incompressible simulations. *HH* is a meshing tool suitable for complex geometries that produces hex-dominant conformal meshes in

* Corresponding author. Tel.: +32-(0)65 37 45 07; fax: +32(0)65 37 45 13
E-mail address: simone.gremmo@umons.ac.be

1877-7058
Peer-review under responsibility of organizing committee of the 24th International Meshing Roundtable (IMR24).

parallel using a *volume-to-surface* approach, including high quality viscous layers starting from unclean geometries. In this research note, the attention will be focused on implementation and results for 2D geometry configurations. For more details about the general 3D geometries, see [12].

2. Strategy Description

The strategy under development can be numbered into the group of partial re-meshing strategies: the region where the mesh needs to be reconstructed should be as limited as possible. Furthermore, it aims at preserving a good mesh quality around the moving bodies in order to assure the solution accuracy. A schematic visualisation of the proposed strategy is given in Fig. 1: in order to follow the movement of the boosters when separating from the shuttle and to minimise the remeshing-time required at each time step, two main concepts are introduced: the **background mesh** is mesh of the computational domain and **moving mesh** is mesh around a moving body that rigidly follows its movement.

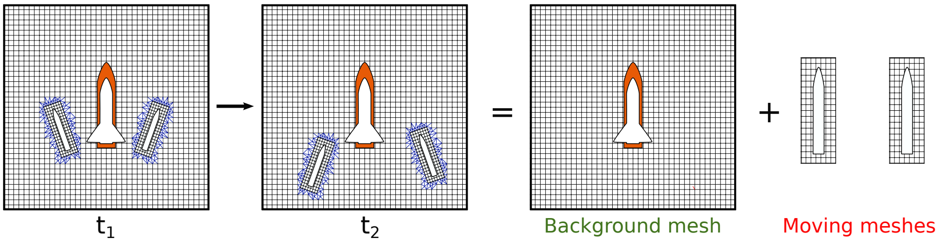

Fig. 1. Partial remeshing strategy: mesh is regenerated only around the moving meshes when the body position is updated during the simulation.

The steps of the algorithm to update the mesh at each new body position are given in Algorithm 1: the crucial idea of the strategy is based on the fact that steps 1 and 2 will be done once at the beginning, so that two reference meshes are obtained; while steps 3 to 6 will be run at each time step to adapt the mesh to new body positions during the simulation. When deleting cells from the background mesh, a hole bigger than the moving mesh is created and to fill this gap, a conformal mesh is generated using *Gmsh* [13] (step 6).

Algorithm 1 Partial re-meshing strategy algorithm

1: Generation of the background mesh
2: Generation of the mesh around each moving object
3: Detection for the overlapping region between meshes
4: Delete cells from background mesh in the overlapping regions
5: Insertion of the moving meshes into the background mesh with holes
6: Generation of a conformal connection between the two meshes.

3. Implementation and Algorithm Description: 2D Configurations

Before starting the description, it is opportune to introduce the nomenclature being used: the **envelope** is the surface delimiting the moving mesh; it may have a generic shape, while the **interface** is the surface delimiting the hole created after removing cells in the overlapping region.

The strategy has also been implemented to handle 2D geometries, specifically all the cases where the body is as large as the domain (e.g. no extremity effects) and the only degrees of freedom are 2 translations in a plane and 1 rotation about an axis perpendicular to the plane; a typical example is an infinitely-long cylinder. It should be underlined that, while the geometrical configuration is 2D, the mesh remains fully three-dimensional. The implementation of the strategy is given in Algorithm 2 is very similar to the implementation for generic 3D geometries ([12]), except that a special care should be taken for the symmetry planes represented by the left and right boundaries. A closed envelope surface is required to detect the overlap and a closed volume have to be provided to *Gmsh* for the remeshing step (see Algorithm 2-step 5). The moving mesh envelope does not define a closed volume and to overcome this issue the extraction of the edge-boundaries delimiting the surface on left and right sides is performed. The lateral taps are created by meshing the two planar surfaces so defined (Algorithm 2-step 2). Analogously, interface and envelope meshes do not bound a closed volume: it is necessary to extract the edge boundaries delimiting both the interface and the envelope and to mesh the planar symmetry surface, creating a "lateral tap" for both right and left sides. A closed volume can now be meshed by *Gmsh* and merged to create the final mesh. As *HH* utilises a volume-to-surface mesh generation approach, it is necessary to interface it with an external mesher, in order to create the conformal connection using a surface-to-volume algorithm. This is mandatory if we want to keep unaltered the faces and vertices belongings to the envelope and the interface.

Algorithm 2 Strategy implementation for 2D geometries

1: Read background-mesh and moving-mesh files
2: Get the envelope of moving mesh label
    Get the right and left edge boundaries on the symmetry planes delimiting the envelope
    Create a closed-volume envelope adding mesh taps on right and left sides
3: Mark vertices in the overlapping region
4: Mark cells that share the vertices marked at step 3 and create the "interface" mesh
    Get the right and left edge boundaries delimiting the interface
5: Create lateral mesh taps on the symmetry planes to delimit the region to be remeshed
6: Remove cells from background mesh
7: Mesh the region delimited by interface and envelope
8: Merge the three meshes and optimise the resulting mesh

4. Results: 2D Cylinder

In the present section results of the mesh partial regeneration process are given for a 2D cylinder configuration. The next example involves only the generation of the mesh without considering the solver side and the focus is set on the time required for the mesh generation as well as on the quality of the resulting mesh.

For this example, we have a 2D cylinder placed in a simple box. Fig. 2 illustrates the reference meshes: the background mesh is given on the left, while the moving mesh is in the middle. The mesh moving rigidly with the body has been built to have a very fine mesh around the solid wall and has been prolonged behind the cylinder to capture its the wake.

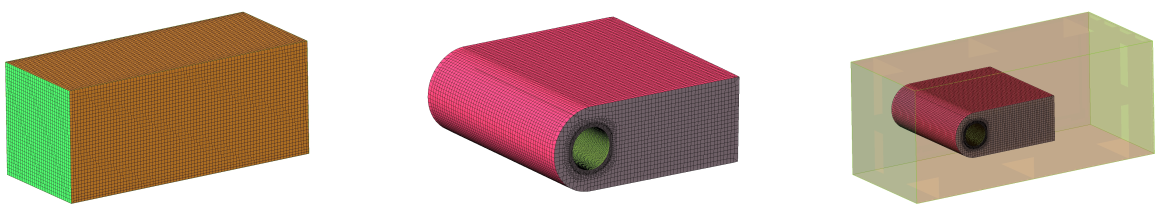

Fig. 2. 2D cylinder case: background mesh (left), moving mesh (middle) and overlapping region (right)

Table 1 resumes the number of cells for the two reference meshes and the time required for their generation. It should be underlined that this is still a "small" case as number of cells is relatively narrow, but still useful to investigate the algorithm robustness and to make preliminary assessments about the time required to run the algorithm described in Algorithm 2. A comparison between a "fine" and a "coarse" test case is provided. The difference between the two test cases lies in the cell characteristic length used for generating the background mesh.

Fig. 3 shows a comparison between the fine (left) and the coarse (right) test cases. In the former case, it can be observed that the remeshed region is a narrow region (yellow mesh) around the moving mesh (in light-green), while for the latter, the moving region is the blue mesh and the remeshed region is a thicker layer in green.

Table 1. Mesh information for 2D cylinder case.

Mesh		Ref. length	# of cells	Time for generation
Background	Fine	150	113780	0 m 23 sec
	Coarse	300	18764	0 m 11 sec
Moving		90	463540	1 m 49 sec

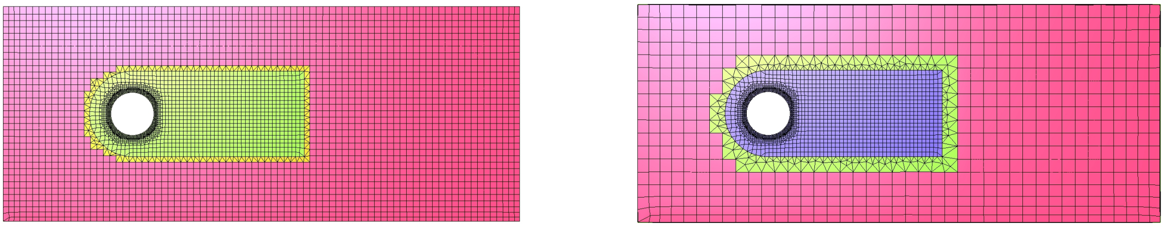

Fig. 3. 2D cylinder case: comparison between resulting meshes for fine (left) and coarse (right) test cases.

Table 2 gives an overview of the resulting meshes for the two test cases. The number of cells added by *Gmsh* for the conformal connection is bigger that the number of cells removed in the overlapping region and the increment is about 14% for the fine test case and 16% for the coarse. Generation time is less than 30 seconds in both cases and includes the mesh optimisation step. It can be observed that the generation time for the coarse mesh is greater and this is explained by the longer time required for improving mesh quality: as the cell size difference between background and moving meshes is bigger, the connecting mesh has initially a lower quality. This shows clearly having a coarser background mesh reduces the mesh regeneration time.

Anyway in both cases the meshes generated have no negative or twisted cells, despite having some bad quality cells (max skewness is less than 0.95 in both cases) in the remeshed region. Fig. 4 and Fig. 5 show that the resulting meshes have acceptable quality and that for coarse mesh the skewness has a worse distribution than for the fine mesh (more elements with skewness close to 1).

Table 2. Generation report for 2D cylinder test case.

2D Cylinder	**Fine Mesh**	**Coarse Mesh**
Total cells	658617	577590
Time	0 m 27 sec	0 m 29 sec

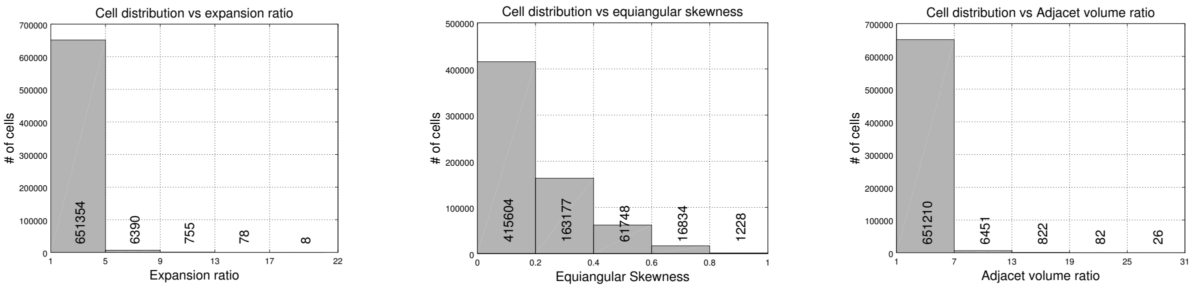

Fig. 4. 2D cylinder fine case: cell distribution VS expansion ratio (left), equiangular skewness (middle) and adjacent volume ratio (right)

5. Conclusion

In this paper a partial remeshing strategy to take into account large displacements has been presented. The implementation for handling 2D geometry configurations is given. The algorithm is based on the idea of generating, at

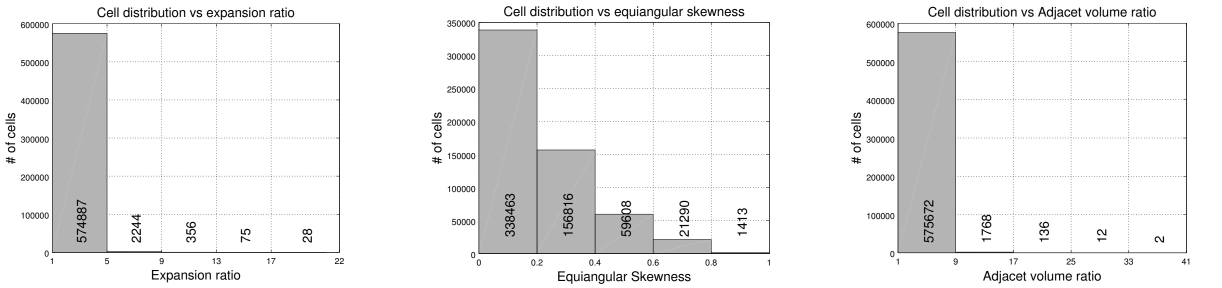

Fig. 5. 2D cylinder coarse case: cell distribution VS expansion ratio (left), equiangular skewness (middle) and adjacent volume ratio (right)

the beginning of the simulation, a high quality mesh around each moving body as well as a background mesh in the computational domain. In order to follow the movement of the bodies, at each time step, it is necessary to detect the overlap between the meshes and successively to create a conformal connection between them. Hence, the remeshed region is a narrow layer and the interpolation affects the solution accuracy only far from the body surfaces. Moreover an ALE formulation can be used for the unmodified moving mesh. The algorithm and implementation are robust enough to generate a valid mesh for all the test cases in acceptable time, including different shapes of the moving mesh envelope and different cell sizes between background and moving meshes. Despite obtaining a valid mesh (e.g. no negative cells), its quality may be improved. It should also be underlined that the number of cells added to create the conformal connection does not depend on the domain size, but rather in the size of the gap between interface and envelope and on the difference of cell size between moving moving mesh and background mesh. Next developments will be dedicated to the optimisation of the gap size between the envelope and the surface in order to improve the quality of the resulting mesh without increasing excessively the mesh generation time. In addition parallel performances will be evaluated for larger meshes.

References

[1] E. E. Panagiotopoulos, S. D. Kyparissis, CFD transonic store separation trajectory predictions with comparison to wind tunnel investigations, International Journal of Engineering 3 (2010) 538–553.

[2] Alex, Cenko, Experience in the use of computational aerodynamics to predict store release characteristics, Progress in Aerospace Sciences 37 (2001) 477 – 495.

[3] A. A. Johnson, T. Tezduyar, Advanced mesh generation and update methods for 3D flow simulations, Computational Mechanics 23 (1999) 130–143.

[4] Zuo-sheng, Yang, Finite element method for the transient process of the separation of external stores from aircraft, Chinese Journal of Aeronautics 15 (2002) 1 – 5.

[5] C. M. Hoke, R. K. Decker, R. M. Cummings, D. R. McDaniel, S. A. Morton, Comparison of overset grid and grid deformation techniques applied to 2-dimensional naca airfoils, in: 19th AIAA Computational Fluid Dynamics, volume AIAA 2009(3537), 2009.

[6] N. C. Prewitt, D. M. Belk, W. Shyy, Parallel computing of overset grids for aerodynamic problems with moving objects, Progress in Aerospace Sciences 36 (2000) 117 – 172.

[7] F. Brezzi, J.-L. Lions, O. Pironneau, Analysis of a chimera method, Comptes Rendus de l'Académie des Sciences - Series I - Mathematics 332 (2001) 655 – 660.

[8] J. A. Witteveen, Explicit and robust inverse distance weighting mesh deformation for cfd, in: 48th AIAA Aerospace Sciences Meeting, volume AIAA 2010(165), 2010.

[9] E. Luke, E. Collins, E. Blades, A fast mesh deformation method using explicit interpolation, Journal of Computational Physics 231 (2012) 586 – 601.

[10] J. Gong, Z. Zhou, B. Liu, Using the unstructured dynamic mesh to simulate multi-store separating from aircraft, Procedia Engineering 16 (2011) 572 – 580. <ce:title>International Workshop on Automobile, Power and Energy Engineering</ce:title>.

[11] H. Jasak, T. zeljko, Dynamic mesh handling in openfoam applied to fluid-structure interaction simulations, in: V European Conference on Computational Fluid Dynamics, 2010.

[12] Gremmo, Simone and Antonik, Iouri and Nigro, Remy and Hirsch, Charles and Coussement, Grégory, Partial remeshing strategy for CFD simulations when large displacements are taken into account, Online, 2014. URL: "http://www.sandia.gov/imr/Papers/IMR23_ResearchNote4_Gremmo.pdf".

[13] C. Geuzaine, J.-F. Remacle, Gmsh: a three dimensional finite element mesh generator with built-in pre- and post-processing facilities, International Journal for Numerical Methods in Engineering 79 (2009) 1309–1331.